Laboratoire de l'Informatique du Parallélisme

École Normale Supérieure de Lyon
Unité Mixte de Recherche CNRS-INRIA-ENS LYON-UCBL n° 5668

Largeur de branche des graphes de cordes

Laurent Lyaudet

Juillet 2004

Rapport de DEA N° 2004-06

École Normale Supérieure de Lyon
46 Allée d'Italie, 69364 Lyon Cedex 07, France
Téléphone : +33(0)4.72.72.80.37
Télécopieur : +33(0)4.72.72.80.80
Adresse électronique : `lip@ens-lyon.fr`

Largeur de branche des graphes de cordes

Laurent Lyaudet

Juillet 2004

Abstract

During this training course, I studied the algorithmic complexity of circle graphs branchwidth computation. I have found an upper bound of this width computable in $O(n^3)$ and an algorithm of linear complexity for resolving, on the intersection of splitgraphs and circle graphs, a problem that is NP-complete on the entire class of splitgraphs.

Keywords : Branchwidth, branch decompositions, circle graphs, splitgraphs

Résumé

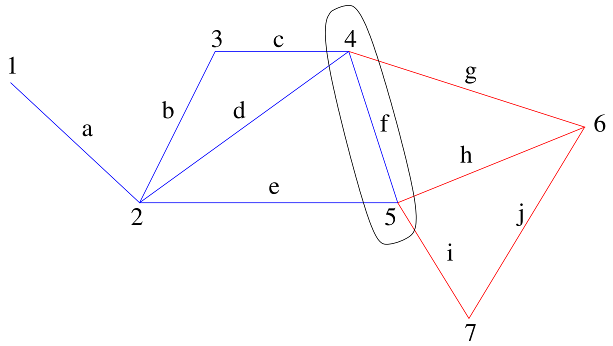
Durant mon stage de DEA, j'ai étudié la complexité algorithmique du calcul de la largeur de branche des graphes de cordes. J'ai obtenu une borne supérieure de cette largeur calculable en $O(n^3)$, ainsi qu'un algorithme de complexité linéaire pour résoudre sur les splitgraphes de cordes un problème NP-complet quand il est considéré sur la classe entière des splitgraphes.

Mots-clés : Largeur de branche, décompositions en branches, graphes de cordes, splitgraphes

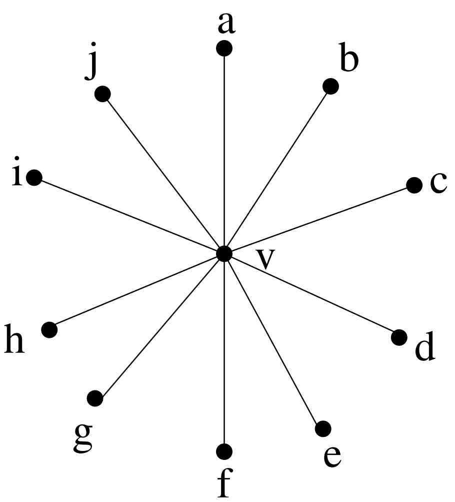
Introduction

Les graphes sont utilisés pour la modélisation de nombreux problèmes d'optimisation. On peut citer comme exemple, les problèmes d'allocation de fréquences qui sont souvent modélisés par des problèmes de coloration de graphes. Partis de considérations purement théoriques, Robertson et Seymour ont développé la théorie des mineurs de graphes afin de démontrer la conjecture de Wagner datant de 1937 : « Dans toute suite infinie de graphes finis, l'un des graphes est mineur d'un autre. » Dans cette théorie, ils ont introduit deux types de décomposition des graphes : la décomposition arborescente et la décomposition en branches. A ces deux types de décompositions sont associées des triangulations des graphes ainsi qu'un invariant : la largeur arborescente et la largeur de branche. Ces deux décompositions sont très liées comme le montre le théorème 5 qui donne une relation min/max entre largeur de branche et largeur arborescente.

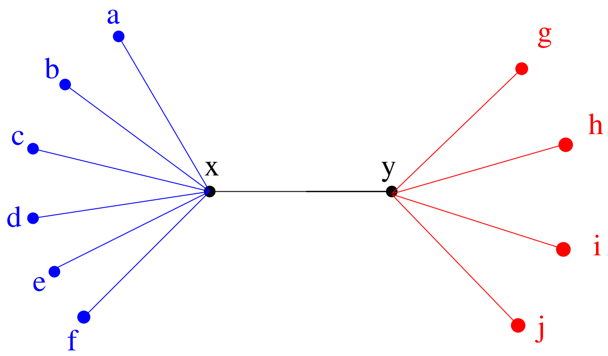
Elles se sont montrées très intéressantes d'un point de vue algorithmique. En effet, elles répondent à la question : « Comment bien diviser pour régner ? » Les décompositions en branches sont basées sur la notion de séparation qui consiste à séparer l'ensemble des arêtes d'un graphe en deux groupes comme le montre le dessin suivant.



Les sommets 4 et 5 représentent la frontière associée à la séparation. On peut voir une décomposition en branches comme l'application récursive de ce principe de séparation. En effet, on peut partir d'un arbre avec un sommet central v dont tous les voisins sont des feuilles étiquetées par les arêtes du graphe.



On éclate alors le noeud v en deux noeuds x et y dont les voisinages correspondent à la séparation.



En appliquant récursivement ce principe d'éclatement des noeuds internes de l'arbre dont le degré est supérieur à 3, on finit par obtenir une décomposition en branches dont la définition formelle est :

Définition 1 (décomposition en branches) *Une* décomposition en branches *d'un graphe* $G = (V, E)$ *est un couple* $\Theta = (T, \tau)$ *où* T *est un arbre ternaire et* τ *est une bijection des feuilles de* T *vers* E*.*

À chaque arête de cette décomposition en branches correspond une séparation des arêtes du graphe. Le « bien diviser » va donc consister à minimiser la taille des frontières associées aux arêtes de la décomposition en branches afin que le coût en temps pour recombiner les solutions partielles soit minimal. Il ne suffit pas d'adopter une démarche gloutonne en éclatant un noeud selon une séparation dont la frontière est minimale pour obtenir une bonne décomposition en branches. En effet, en suivant cette heuristique gloutonne, on peut obtenir un résultat arbitrairement éloigné de la valeur optimale.

Courcelle [5] a montré que tous les problèmes de décision pouvant s'exprimer dans la logique monadique du second ordre sont décidables en temps linéaire pour une classe de graphes ayant une largeur arborescente ou une largeur de branche bornée. Parmi ces problèmes, figurent des problèmes NP-complets comme l'existence d'un circuit hamiltonien, la taille d'un stable maximum ou le nombre chromatique [1]. Néanmoins les algorithmes obtenus sont avec une constante exponentielle en la largeur qui borne la classe.

Cook et Seymour [4] ont aussi utilisé les décompositions en branche pour créer l'une des meilleurs heuristiques pour le problème du voyageur de commerce.

Le calcul de la largeur arborescente ou de la largeur de branche sont deux problèmes NP-complets. Néanmoins, ces problèmes sont polynomiaux lorsqu'on se restreint à certaines classes de graphes. De ce point de vue, la largeur arborescente a été beaucoup plus étudiée que la largeur de branche. Cela est dû notamment à l'antériorité de la largeur arborescente qui a été introduite en 1984 dans [14], alors que la largeur de branche n'a été introduite que 7 ans plus tard dans [15]. On sait que le calcul de la largeur arborescente est polynomial pour les graphes de cordes, les graphes d'intervalles et d'intervalles circulaires. De plus, les méthodes utilisées pour la calculer sont quasiment identiques pour ces classes de graphes. Ces trois classes de graphes sont des classes de graphes d'intersection. Les graphes d'intersection ont souvent une structure géométrique qui leur donne des propriétés topologiques. Ces propriétés les rendent souvent plus faciles à manipuler et représentent parfois des propriétés rencontrées dans les modélisations de problème d'optimisation.

Les graphes d'intervalles circulaires servent, par exemple, en compilation pour représenter la durée de vie d'une famille de variables dans une boucle et une coloration du graphe correspond à une répartition optimale des variables dans les registres.

Définition 2 (graphe d'intersection) *Soit $\mathcal{F}$ une famille non vide d'ensembles, on lui associe le* graphe d'intersection $G_{\mathcal{F}}$ *défini comme suit :*
- *à tout $X \in \mathcal{F}$, on associe un sommet x ;*
- *$xy \in E$ si $X \cap Y \neq \emptyset$.*

En considérant des familles d'objets particuliers, on obtient des classes de graphes avec des propriétés intéressantes, comme les graphes de cordes, les graphes d'intervalles et d'intervalles circulaires.

Définition 3 (graphe de cordes, modèle circulaire) *Un* graphe de cordes *est le graphe d'intersection de cordes d'un cercle. Un cercle et un ensemble de cordes forment un* modèle circulaire *du graphe de cordes correspondant.*

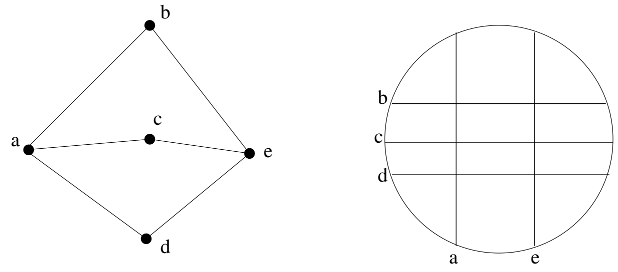


FIG. 1 – Un graphe de cordes et son modèle circulaire

Définition 4 (graphe d'intervalles circulaires) *Un* graphe d'intervalles circulaires *est le graphe d'intersection d'arcs d'un cercle. Un cercle et un ensemble d'arcs forment un modèle circulaire du graphe d'intervalles circulaires correspondant.*

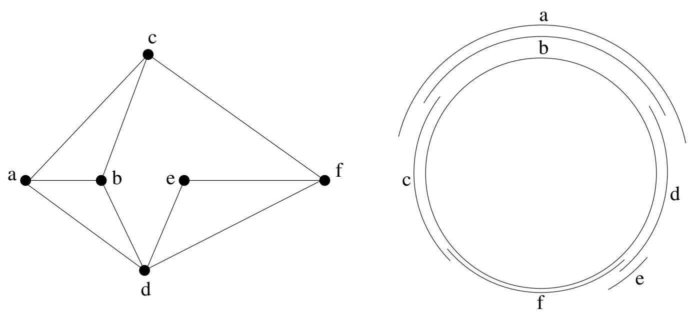


FIG. 2 – Un graphe d'intervalles circulaires et son modèle circulaire

Définition 5 (graphe d'intervalles) *Un* graphe d'intervalles *est le graphe d'intersection de segments de la droite réelle.*

Kloks et al [11] ont montré que le calcul de la largeur de branche des graphes d'intervalles est polynomial.

Mon sujet de stage était initialement de modifier légérement la méthode que Kloks [10] avait utilisée pour calculer la largeur arborescente des graphes de cordes afin de calculer la largeur de branche de ces graphes. Du fait des liens entre décompositions arborescentes et décompositions en branches, il semblait raisonnable d'espérer un résultat positif pour les graphes de cordes. D'ailleurs, Mazoit [8] a montré, pendant mon stage, qu'une méthode proche pour les graphes d'intervalles circulaires s'adaptait bien.

La classe des graphes de cordes est relativement bien étudiée d'un point de vue algorithmique. Tout d'abord, l'appartenance ou non à cette classe est polynomiale. Le meilleur algorithme actuel étant un algorithme de complexité quadratique dû à Spinrad ([17]). Cet algorithme fournit aussi un modèle circulaire lorsque le graphe est un graphe de cordes. Un certain nombre de problèmes NP-complets, comme la recherche d'un indépendant de poids maximal ou d'une clique de poids maximal ([7, 9, 16]), peuvent être résolus en temps polynomial pour les graphes de cordes. Mais certains restent NP-complets, comme le problème du nombre chromatique [12] ou celui de l'existence d'un circuit hamiltonien [6].

Dans une première section, je vais présenter des définitions et des résultats du domaine nécessaires à la compréhension du rapport. Puis je présenterai une tentative de réduction vers un problème NP-complet pour les splitgraphes de cordes et je donnerai un algorithme linéaire pour décider si la largeur de branche d'un splitgraphe de corde est égale à $\lceil \frac{2}{3}|C| \rceil$. J'essaierai ensuite d'appliquer la méthode utilisée par Kloks pour calculer la largeur arborescente des graphes de cordes à la largeur de branche. Ce qui me founira une borne inférieure et une borne supérieure de cette largeur sans que j'arrive à faire coincider les deux bornes. J'essaierai enfin d'appliquer la méthode utilisée par Mazoit [8] pour calculer la largeur de branche des graphes de cordes avant de conclure.

0.1 Quelques définitions

0.1.1 Décompositions arborescentes et triangulations

Les définitions et résultats qui suivent sont relatifs aux triangulations et à la largeur arborescente et sont nécessaires à la compréhension des sections suivantes. Leurs démonstrations sont présentées de manière claire dans [18].

Définition 6 (graphe triangulé, corde) *Un graphe est dit* triangulé *si tout cycle de longueur supérieure ou égale à 4 admet une* corde*, c'est-à-dire une arête entre deux sommets non consécutifs du cycle. Ces cordes n'ont rien à voir avec les cordes des graphes de cordes.*

Définition 7 (sous-graphe induit) *Soient $G = (V, E)$ un graphe et $V' \subseteq V$. Le* sous-graphe *de G* induit *par V', noté $G[V']$, est le sous-graphe (V', E') où $E' = \{v_1v_2 \in E / v_1, v_2 \in V'\}$.*

Proposition 1 *Tout sous-graphe induit d'un graphe triangulé est triangulé.*

Définition 8 (séparateur minimal) *Soit $G = (V, E)$ un graphe, $S \subseteq V$ est un a, b-*séparateur *$(a, b \in V)$ si a et b sont dans des composantes connexes différentes de $G \backslash S$. S est un a, b-*séparateur minimal *si S est un a, b-séparateur et si aucun sous-ensemble strict de S n'est un a, b-séparateur. S est un* séparateur minimal *s'il existe $a, b \in V$ tels que S soit un a,b-séparateur minimal.*

Théorème 1 (Dirac,1961) *Un graphe $G = (V, E)$ est triangulé si et seulement si tous ses séparateurs minimaux sont des cliques.*

Définition 9 (arbre de cliques) *Soit $G = (V, E)$ un graphe, $T = (\mathcal{K}, \mathcal{E})$ est un* arbre de cliques *de G si :*

- *$\mathcal{K}$ est l'ensemble des cliques maximales de G (pour $\subseteq$) ;*
- *$\forall v \in V$, $T[K_v]$ est un sous-arbre de T où K_v est l'ensemble des cliques maximales contenant v.*

Théorème 2 *Soit $G = (V, E)$ un graphe, il y a équivalence entre :*

- *G est triangulé ;*
- *G est le graphe d'intersection des sous-arbres d'un arbre ;*
- *G admet un arbre de cliques.*

Théorème 3 *Soit G un graphe triangulé et $\mathcal{T}$ un arbre de clique de G, S est un séparateur minimal de G si et seulement si $S = \Omega \cap \Omega'$ où Ω et Ω' sont deux cliques maximales de G adjacentes dans $\mathcal{T}$.*

Définition 10 (triangulation) *Soit $G = (V, E)$ un graphe quelconque. On dit que le graphe $H = (W, F)$ est une* triangulation *de G si $W = V$, $E \subseteq F$ et H est triangulé. On dit que $H = (V, F)$ est une* triangulation minimale *de $G = (V, E)$ si $\forall F'$ $E \subseteq F' \subset F$ le graphe (V, F') n'est pas triangulé.*

Définition 11 (décomposition arborescente, largeur arborescente)
Une décomposition arborescente *d'un graphe $G = (V, E)$ est un couple $(T = (I, E_T), \{X_i, i \in I\})$ où T est un arbre et les X_i sont des ensembles de sommets de G qui étiquettent les sommets de T tels que :*

- *$\bigcup_{i \in I} X_i = V$;*
- *$\forall xy \in E, \exists i \in I$ tel que $x, y \in X_i$;*
- *$\forall x \in V$, le sous-graphe induit par l'ensemble T_x des sommets $i \in I$ tels que $x \in X_i$ est un sous-arbre de T.*

La largeur d'une décomposition est le $\max_{i \in I} |X_i| - 1$. La largeur arborescente *d'un graphe, notée $\alpha(G)$, est le minimum des largeurs de ses décompositions.*

Le -1 dans la formule de la largeur a été introduit pour que la largeur arborescente d'un arbre ou d'une forêt soit égale à 1.

Les décompositions arborescentes et les triangulations d'un graphe sont fortement liées. Considérons un graphe G et une décomposition arborescente T de G. Si on ajoute au graphe G des arêtes afin que les sous-graphes induits par chaque étiquette de T soient des cliques, alors on obtient une triangulation H de G et T est un arbre des cliques de H.

Ainsi, pour trouver une bonne décomposition arborescente de G, il suffit de chercher une bonne triangulation H de G, c'est-à-dire une triangulation avec la clique maximum la plus petite possible. Or si $G \subseteq H' \subset H$ est une triangulation de G, les cliques de H' sont contenues dans des cliques de H, la triangulation H' n'est pas moins bonne que H. On peut donc se restreindre aux triangulations minimales.

Lemme 1 *Soient $G = (V, E)$ un graphe, $K \subseteq V$ une clique de G et $(T = (I, E_T), \{X_i, i \in I\})$ une décomposition arborescente de G. Alors il existe $i \in I$ tel que $K \subseteq X_i$.*

Corollaire 1 *Soit K_n la clique à n sommets, $\alpha(K_n) = n - 1$*

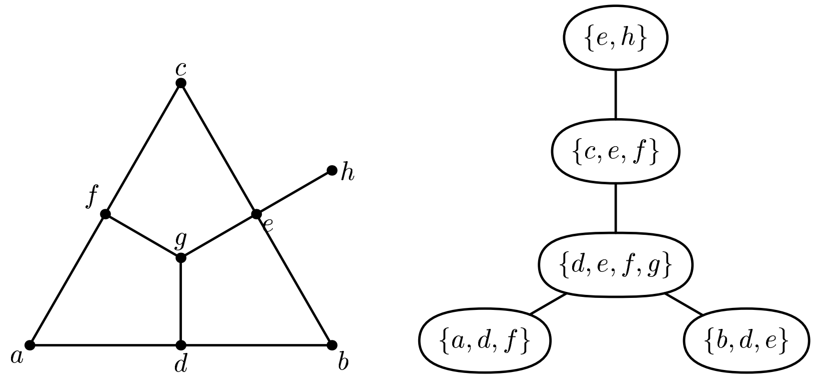


Fig. 3 – Un graphe et une décomposition arborescente associée

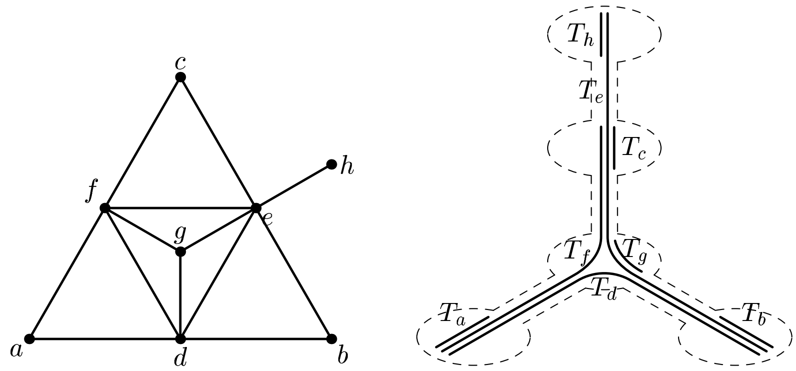


Fig. 4 – La triangulation correspondante et son modèle d'intersection

Proposition 2 *La largeur arborescente de G est égale au minimum, pour toutes les triangulations H de G, de $\omega(H) - 1$ où $\omega(H)$ est la taille de la plus grande clique de H.*

Définition 12 *Soient un graphe $G = (V, E)$ et $S, T \subseteq V$. On dit que S croise T et on le note $S\sharp T$ si S intersecte au moins deux composantes connexes de $G\backslash T$. Dans le cas contraire, on dit que S est parallèle à T, ce que l'on note $S||T$. Ces deux relations sont symétriques.*

Les séparateurs minimaux jouent un rôle important dans l'étude des décompositions arborescentes. Le théorème suivant dû à Parra et Scheffler montre qu'ils suffisent à caractériser les triangulations minimales et donc les « bonnes » décompositions arborescentes d'un graphe.

Théorème 4 ([13]) *Soient $G = (V, E)$ un graphe, Γ une famille maximale de séparateurs minimaux deux à deux parallèles de G et H le graphe obtenu en ajoutant à G des arêtes pour que les séparateurs minimaux de Γ soient des cliques. Alors, le graphe H est une triangulation minimale de G et l'ensemble de ses séparateurs minimaux est Γ.*

Inversement, si H est une de ses triangulations minimales. L'ensemble Δ_H des séparateurs minimaux de H est une famille maximale de séparateurs minimaux de G deux à deux parallèles et H est obtenu en complétant les séparateurs minimaux de Δ_H.

0.1.2 Définitions et généralités sur la largeur de branche

Cette section reprend plus en détails les idées et résultats relatifs aux décompositions en branche présentées dans l'introduction. Ils sont tirés de l'article [15] de Robertson et Seymour.

Définition 13 (séparation, frontière) *Une* séparation Σ *d'un graphe G est une bipartition de $E(G)$ en E_1 et E_2. Soit G_1 (respectivement G_2) l'ensemble des sommets incidents à au moins une arête dans E_1 (respectivement E_2). On définit $F(\Sigma) = G_1 \cap G_2$ comme étant la* frontière *associée à Σ.*

Définition 14 (ordre d'une arête) *L'*ordre *d'une arête e de T est le nombre de sommets v de G tels qu'il existe des feuilles t_1, t_2 de T dans différentes composantes de $T\backslash e$, avec $\tau(t_1)$ et $\tau(t_2)$ incidentes à v. C'est donc la taille de la frontière $F(e)$ associée à la séparation de G induite par e.*

Définition 15 (largeur de branche) *La largeur de $\Theta = (T, \tau)$ est l'ordre maximum des arêtes de T. La* largeur de branche *de G, notée $\beta(G)$, est la largeur minimum des décompositions en branches de G, ou 0 si G n'a pas de décomposition en branches (si $|E(G)| \leq 1$) .*

Définition 16 (mineur) *Le graphe H est un* mineur *du graphe G, s'il peut être obtenu à partir de G en supprimant une arête ou un sommet ou en contractant une arête. Contracter une arête consiste à remplacer ses deux extrémité par un seul sommet ayant pour voisinage l'union des voisinages des extrémités.*

Propriété 1 ([15]) *Si H est un mineur d'un graphe G, alors $\beta(H) \leq \beta(G)$.*

Propriété 2 ([15]) *Un graphe G a une largeur de branche :*

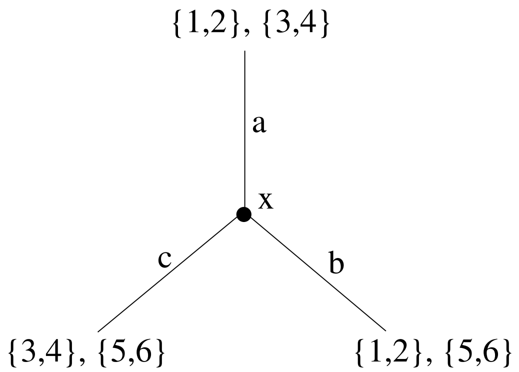
i) égale à 0 si et seulement si toute composante connexe de G a au plus une arête ;

ii) inférieure ou égale à 1 si et seulement si toute composante connexe a au plus un sommet de degré supérieur ou égal à 2 ;

iii) inférieure ou égale à 2 si et seulement si G n'a pas K_4 comme mineur.

Propriété 3 ([15]) *Pour $n \geq 3$, $\beta(K_n) = \lceil \frac{2n}{3} \rceil$.*

En effet, pour construire une décomposition en branches $\Theta = (T, \tau)$ de largeur minimale de K_n, on partage l'ensemble des sommets en 3 groupes de couleurs différentes de taille au plus $\lceil \frac{n}{3} \rceil$. On va, par exemple, partager les sommets de K_6 en 3 groupes $C_1 = \{1, 2\}, C_2 = \{3, 4\}$ et $C_3 = \{5, 6\}$. Puis on va répartir les arêtes sur 3 branches comme le montre le schéma suivant.



Les arêtes entre des sommets de C_1 et C_2 vont se retrouver dans la branche a, celles entre des sommets de C_1 et C_3 vont se retrouver dans la branche b et celles entre des sommets de C_2 et C_3 vont se retrouver dans la branche c. Toutes les arêtes de T dans les branches a, b ou c seront d'ordre au plus $\lceil \frac{2n}{3} \rceil$. Ce sera également le cas pour les arêtes de T qui joignent le sommet central x à une branche.

Cette idée de 3-coloration est importante pour la partie qui suit sur les splitgraphes de cordes.

Comme on l'a déjà mentionné, les décompositions arborescentes et les décompositions en branches sont très liées. Le théorème suivant l'illustre bien. Nous ne donnerons pas la preuve ici mais celle-ci utilise de façon plus poussée ces liens. Elle transforme une décomposition en branche en une décomposition arborescente et inversement et chacune de ces deux transformations donne l'une des deux inégalitées.

Théorème 5 ([15]) *Soit G un graphe tel que $\beta(G) \geq 2$, on a alors :*

$$\beta(G) \leq \alpha(G) + 1 \leq \left\lfloor \frac{3}{2}\beta(G) \right\rfloor$$

0.2 Les splitgraphes de cordes

Cette partie est dédiée à la preuve du théorème 6 qui fournit un algorithme linéaire qui décide si la largeur de branche d'un splitgraphe de cordes vaut $\lceil \frac{2}{3}|C| \rceil$.

Définition 17 *Un graphe $G = (V, E)$ est un splitgraphe si l'on peut partager V en un indépendant I de G (ensemble de sommets non adjacents deux à deux) et une clique C de G. On note parfois $G = (I, C, E)$.*

Kloks, Kratochvil et Müller ont montré [11] que le calcul de la largeur de branche d'un splitgraphe était NP-complet. Sauf si P=NP, ce résultat prouve que la complexité du calcul de la largeur de branche et celle du calcul de la largeur arborescente peuvent être totalement différents sur certaines classes de graphes. En effet, les splitgraphes forment une sous-classe des graphes triangulés et donc leur largeur arborescente est obtenue en calculant la taille de leur plus grande clique qui est égale à $|C|$ ou à $|C|+1$ si l'un des sommets de I est adjacent à tous les sommets de C. Le résultat de NP-complétude du calcul de la largeur de branche d'un splitgraphe est basée sur le lemme suivant qui reprend l'idée de 3-coloration pour obtenir une décomposition en branches optimale d'une clique. On cherche à isoler chaque sommet de l'indépendant dans l'une des 3 branches.

Lemme 2 *Soit $G = (I, C, E)$ un splitgraphe. C étant une clique de G, $\beta(G) \geq \lceil \frac{2}{3}|C| \rceil$. De plus, l'égalité n'est vérifiée que s'il existe une partition de C en trois ensembles C_1,C_2 et C_3 tels que $|C_i| \leq \lceil \frac{1}{3}|C| \rceil$ et pour tout sommet $i \in I$, il existe $j \in \{1, 2, 3\}$ tel que $N(i) \cap C_j = \emptyset$ où $N(i)$ est le voisinage du sommet i. Lorsqu'une telle partition de C existe, on dira qu'il existe une 3-coloration acceptable de C dans G.*

Kloks et al exhibent ensuite une réduction de ce problème vers celui de la 3-colorabilité des graphes. L'intersection des splitgraphes et des graphes de cordes n'étant ni vide ni triviale, j'ai cherché une réduction à peu près similaire pour montrer que le calcul de la largeur de branche des splitgraphes de cordes était NP-complet.

Mais à la place, j'ai exhibé un algorithme linéaire qui décide si la largeur de branches d'un splitgraphe de corde $G = (I, C, E)$ est égale à $\lceil \frac{2}{3}|C| \rceil$.

Ce résultat montre que l'approche de Kloks et al ne marche pas pour les splitgraphes de corde. On peut donc supposer que le problème de la largeur de branches est plus simple dans le cas des graphes de cordes. Cependant, mon algorithme ne résoud le problème que pour une valeur bien précise. Il ne permet pas de calculer la valeur de la largeur de branche d'un splitgraphe de cordes.

Avant de continuer plus avant, nous allons montrer que nous pouvons restreindre la forme des splitgraphes de cordes. Comme nous l'avons déjà remarqué, une fois que nous avons choisi une partition C_1, C_2, C_3 de la clique, on regarde si le voisinage des sommets de l'indépendant ne rencontre que deux des trois ensembles C_i. Dans cette optique, seuls les sommets de I dont le voisinage est maximal sont intéressants.

Lemme 3 *Soient $G = (I, C, E)$ un splitgraphe et $G' = (I', C, E)$ le splitgraphe obtenu à partir de G en enlevant d'abord les sommets de I dont le voisinage est de taille inférieure ou égale à 2 ou est strictement inclus dans le voisinage d'un autre sommet de I puis en ne gardant qu'un sommet de l'indépendant parmi ceux qui ont le même voisinage. G admet une 3-coloration acceptable de C si et seulement si G' admet une 3-coloration acceptable de C.*

Preuve :

$\Rightarrow$: Puisque G admet une 3-coloration acceptable de C, tous les sommets de I donc de I' acceptent cette coloration, et donc G' admet une 3-coloration acceptable de C.

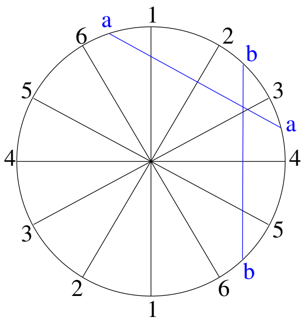
$\Leftarrow$: Les sommets de I dont le voisinage est de cardinal inférieur ou égal à 2 acceptent toute 3-coloration de C. Les autres sommets de I ont tous leur voisinage inclus dans celui d'un sommet de I', donc s'il existe une 3-coloration acceptable de C dans G', tous les sommets de I' l'acceptent et donc ceux de I l'acceptent aussi. D'où G admet une 3-coloration acceptable de C.

■

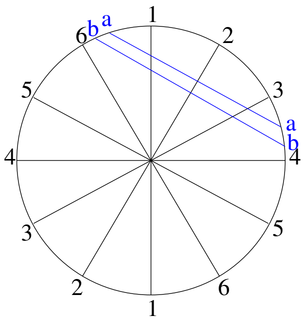
Dans la suite, on considérera toujours un splitgraphe de cordes (I, C, E) dont les sommets de I vérifient les conditions de sélection du lemme. Soit $M(G)$ un modèle circulaire de G.

Soient deux sommets a et b de I représentés par des cordes bleues. Les configurations suivantes sont impossibles :

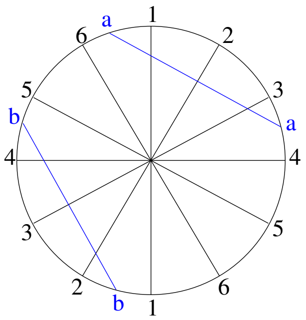
– deux cordes qui se croisent car sinon a et b sont adjacents



– deux cordes croisant une même corde de C du même côté du centre du cercle car sinon $N(a) \subseteq N(b)$ ou $N(b) \subseteq N(a)$



Chaque sommet de C est donc adjacent à au plus deux sommets de I.



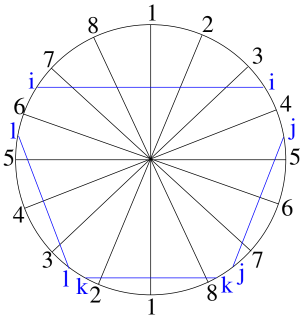
Ceci nous conduit à définir la notion de pseudo-voisins.

Définition 18 (pseudo-voisins) *Soit $G = (I, C, E)$ un splitgraphe, deux sommets a, b de I sont dits* pseudo-voisins *s'il existe un sommet c de C tel que a, b sont tous deux adjacents à c.*

Lemme 4 *Soit $G = (I, C, E)$ un splitgraphe de corde où I vérifie les conditions du lemme 3 alors tout sommet de I a au plus deux pseudo-voisins dans I.*

Preuve :

Supposons qu'il existe $i, j, k, l \in I$ tels que j, k et l soient tous des pseudo-voisins de i. Soit $M(G)$ un modèle circulaire de G. Supposons sans perte de généralité, que l'on rencontre, en parcourant le cercle dans le sens des aiguilles d'une montre et en partant d'une extrémité de la corde correspondant à i, les cordes correspondant aux sommets j, k, l dans cet ordre.



On a alors que $N(k) \subseteq N(i)$ ce qui contredit les conditions du lemme 3. ■

Corollaire 2 *Tout sommet de I appartient à un unique chemin maximal ou à un unique cycle de pseudo-voisins dans I.*

Proposition 3 *S'il existe un cycle de pseudo-voisins $Cy \subseteq I$ alors son voisinage vérifie $N(Cy) = C$ et $Cy = I$.*

Preuve :

Soit $Cy = (c_1, c_2, \ldots, c_m)$, $m \geq 3$, un cycle de pseudo-voisins. On numérote les sommets de C de 1 à $n = |C|$, en partant de la première corde d'un sommet de C dans $M(G)$ intersectée par la corde correspondant à c_1 et en suivant le cercle dans le sens des aiguilles d'une montre. $N(c_i)$ peut donc être vu comme une suite continue croissante de C en considérant que le passage de n à 1 est continu et croissant. La corde correspondant à c_1 intersecte donc les cordes des sommets $1, \ldots, i_1$. Les deux pseudo-voisins de c_1 sont c_2 et c_m. Supposons, sans perte de généralité, que $N(c_2)$ (respectivement $N(c_m)$) intersecte $N(c_1)$ par la droite (respectivement par la gauche), c'est à dire que $N(c_2) = i_1 - k_1, \ldots, i_2$, avec $0 \leq k_1 \leq i_1 - 2$ et $i_2 > i_1$, et $N(c_m) = i_{m-1} - k_{m-1}, \ldots, n, 1, i_m$, avec $0 \leq k_m - 1 \leq i_m - 2$ et $1 \leq i_m < i_1 - k_1$.

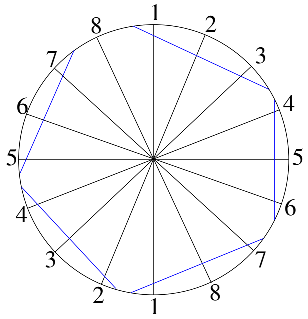
Les $N(c_i)$ vont donc être des sous-intervalles de $[1..n]$ sauf $N(c_m)$ qui est l'union des sous-intervalles $[i_{m-1} - k_{m-1}..n]$ et $[1..i_m]$. De plus ces sous-intervalles recouvrent $[1..n]$. On a donc bien

$$N(Cy) = \bigcup_{1 \leq i \leq m} N(c_i) = C$$

et en conséquence $I = Cy$. ■

On peut donc considérer que I est soit un cycle de pseudo-voisins, soit l'union disjointe de chemins maximaux de pseudo-voisins.

Avant de passer à la proposition 4, nous allons maintenant donner deux conditions qui interdisent l'existance d'une 3-coloration acceptable de C dans G. S'il existe un sommet i de I tel que $|N(i)| > \left\lceil \frac{2}{3}|C| \right\rceil$ alors il n'existe pas de 3-coloration acceptable de C dans G. On peut donc supposer que $\forall i \in I,\ 3 \leq |N(i)| \leq \left\lceil \frac{2}{3}|C| \right\rceil$. De plus, on a $|I| \leq \left\lfloor \frac{2}{3}|C| \right\rfloor$ car il y a $2|C|$ extrémités des cordes de C et chaque corde de I couvre au moins trois extrémités. Cette borne est atteinte, comme le montre l'exemple suivant.



Lemme 5 (conflit) *S'il existe* i *et* j *dans* I *tels que* $|N(i) \cap N(j)| > \lceil \frac{1}{3}|C| \rceil$ *et* $|N(i) \cup N(j)| > \lceil \frac{2}{3}|C| \rceil$ *alors il n'existe pas de 3-coloration acceptable de* C *dans* G*. Dans ce cas, on dira que* i *et* j *sont* en conflit*. De plus, si* i *et* j *sont les seuls sommets de* I*, alors* $|N(i) \cap N(j)| > \lceil \frac{1}{3}|C| \rceil$ *et* $|N(i) \cup N(j)| > \lceil \frac{2}{3}|C| \rceil$ *si et seulement si il n'existe pas de 3-coloration acceptable de* C *dans* G*.*

Preuve :

$\Rightarrow$: $|N(i) \cap N(j)| > \lceil \frac{1}{3}|C| \rceil$ donc les sommets de C dans $|N(i) \cap N(j)|$ doivent être colorés avec deux couleurs donc tous les sommets de $|N(i) \cup N(j)|$ doivent être colorés avec ces deux couleurs. Il est donc impossible que G admette une 3-coloration acceptable de C puisque deux couleurs ne doivent pas colorer plus de $\lceil \frac{2}{3}|C| \rceil$ sommets.

$\Leftarrow$: Raisonnement par contraposée : si i et j sont les seuls sommets de I, et $|N(i) \cap N(j)| \leq \lceil \frac{1}{3}|C| \rceil$ ou $|N(i) \cup N(j)| \leq \lceil \frac{2}{3}|C| \rceil$ alors il existe une 3-coloration acceptable de C dans G.

Premier cas : si $|N(i) \cap N(j)| \leq \lceil \frac{1}{3}|C| \rceil$ alors on peut colorer $N(i) \cap N(j)$ avec une unique couleur, la couleur 1. On peut colorer $N(i) \backslash N(j)$ par la couleur 2, puis la couleur 1 si $|N(i) \backslash N(j)| > \lceil \frac{1}{3}|C| \rceil$, puisque $|N(i)| \leq \lceil \frac{2}{3}|C| \rceil$. De même, on peut colorer $N(j) \backslash N(i)$ par la couleur 3, puis la couleur 1 si $|N(j) \backslash N(i)| > \lceil \frac{1}{3}|C| \rceil$, puisque $|N(j)| \leq \lceil \frac{2}{3}|C| \rceil$. Au final, la couleur 1 colorera au maximum $|N(i) \cap N(j)| + |N(i) \backslash N(j)| - \lceil \frac{1}{3}|C| \rceil + |N(j) \backslash N(i)| - \lceil \frac{1}{3}|C| \rceil \leq |C| - \lceil \frac{1}{3}|C| \rceil - \lceil \frac{1}{3}|C| \rceil = \lceil \frac{1}{3}|C| \rceil$.

Deuxième cas : si $|N(i) \cup N(j)| \leq \lceil \frac{2}{3}|C| \rceil$ alors on peut colorer $|N(i) \cup N(j)|$ avec les deux couleurs de notre choix, et on colore le reste de C sans aucune restriction.

On a donc une 3-coloration acceptable de C dans les deux cas. ■

En fait, les deux propositions suivantes montrent que ces deux conditions sont les deux seules interdisant l'existence d'une 3-coloration acceptable.

Proposition 4 *Si* I *se partage en un ensemble de chemins maximaux de pseudo-voisins et qu'aucun couple de ses sommets n'est en conflit alors il existe une 3-coloration acceptable de* C *dans* G*.*

Preuve :

Les différents chemins maximaux ont des voisinages disjoints dans C.

<u>1er cas</u> : Supposons qu'il n'existe pas de chemin maximal dont le voisinage soit plus grand que $\lceil \frac{2}{3}|C| \rceil$, alors il existe au plus deux chemins maximaux Ch_1 et Ch_2 tels que $|N(Ch_1)| > \lceil \frac{1}{3}|C| \rceil$ et $|N(Ch_2)| > \lceil \frac{1}{3}|C| \rceil$.

<u>1er sous-cas</u> : S'il en existe au moins 1, on colore les sommets de $N(Ch_1)$ avec la couleur 1 puis quand on a épuisé le quota, on passe à la couleur 2. Comme $|N(Ch_1)| \leq \lceil \frac{2}{3}|C| \rceil$, les deux couleurs 1 et 2 suffisent à colorer $N(Ch_1)$. De plus, on a épuisé la couleur 1 donc en colorant

comme on veut le reste de C avec les couleurs 2 et 3, on est sûr d'avoir une 3-coloration acceptable de C dans G.

2e sous-cas : S'il n'existe pas de chemin maximal Ch_1 dont le voisinage vérifie $|N(Ch_1)| > \lceil \frac{1}{3}|C| \rceil$, il suffit de choisir la couleur 1, de colorer le voisinage d'un chemin maximal avec ; de continuer avec un autre chemin maximal, jusqu'à épuisement du quota de 1, puis de passer à la couleur 2 et à la couleur 3. En procédant ainsi, on est sûr qu'aucun chemin maximal n'aura plus de deux couleurs dans son voisinage puisqu'il faut colorer au moins $\lceil \frac{1}{3}|C| \rceil + 2$ sommets pour avoir deux changements de couleur. On obtient ainsi une 3-coloration acceptable de C dans G.

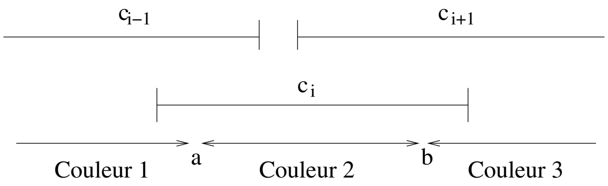
2e cas : Supposons qu'il existe un chemin maximal Ch_1 dont le voisinage vérifie $|N(Ch_1)| > \lceil \frac{2}{3}|C| \rceil$ et $Ch_1 = (c_1, \ldots, c_m)$, $m \geq 2$. Si $m = 2$, comme il n'existe pas de conflits, on sait que $|N(c_1) \cap N(c_2)| \leq \lceil \frac{1}{3}|C| \rceil$. On se retrouve dans le premier cas du lemme 5. On va modifier la coloration afin de s'assurer qu'on épuise au moins l'une des 3 couleurs et donc que l'on pourra colorer les autres chemins maximaux avec la ou les couleurs restantes sans restrictions. On peut colorer $N(c_1) \cap N(c_2)$ avec une unique couleur, la couleur 1. On peut colorer $N(c_1) \backslash N(c_2)$ par la couleur 2, puis la couleur 1 si $|N(c_1) \backslash N(c_2)| > \lceil \frac{1}{3}|C| \rceil$, puisque $|N(c_1)| \leq \lceil \frac{2}{3}|C| \rceil$. De même, on peut colorer $N(c_2) \backslash N(c_1)$ par la couleur 3, puis la couleur 1 si $|N(c_2) \backslash N(c_1)| > \lceil \frac{1}{3}|C| \rceil$, puisque $|N(c_2)| \leq \lceil \frac{2}{3}|C| \rceil$. Au final, la couleur 1 colorera au maximum $|N(c_1) \cap N(c_2)| + |N(c_1) \backslash N(c_2)| - \lceil \frac{1}{3}|C| \rceil + |N(c_2) \backslash N(c_1)| - \lceil \frac{1}{3}|C| \rceil \leq |C| - \lceil \frac{1}{3}|C| \rceil - \lceil \frac{1}{3}|C| \rceil = \lceil \frac{1}{3}|C| \rceil$. Si pour colorer $N(c_1) \backslash N(c_2)$ ou $N(c_2) \backslash N(c_1)$, on a été obligé de revenir à la couleur 1, c'est qu'on a épuisé la couleur 2 ou la couleur 3.

Sinon, cela signifie que $N(c_1) \backslash N(c_2) \leq \lceil \frac{1}{3}|C| \rceil$ et $N(c_2) \backslash N(c_1) \leq \lceil \frac{1}{3}|C| \rceil$. De plus, comme $|N(c_1) \cup N(c_2)| > \lceil \frac{2}{3}|C| \rceil$, on a $|N(c_1)| > \lceil \frac{1}{3}|C| \rceil$ ou $|N(c_2)| > \lceil \frac{1}{3}|C| \rceil$. Supposons, sans perte de généralité, que $|N(c_1)| > \lceil \frac{1}{3}|C| \rceil$. On peut donc épuiser la couleur 1 en colorant d'abord $N(c_1) \cap N(c_2)$, puis $N(c_1) \backslash N(c_2)$ et finir $N(c_1) \backslash N(c_2)$ avec la couleur 2 puisque $N(c_1) \backslash N(c_2) \leq \lceil \frac{1}{3}|C| \rceil$. On peut toujours colorer $N(c_2) \backslash N(c_1)$ avec la couleur 3, puisque $N(c_2) \backslash N(c_1) \leq \lceil \frac{1}{3}|C| \rceil$. On obtient donc une 3-coloration acceptable de C dans G.

On suppose maintenant que $m \geq 3$.

1er sous-cas : S'il n'existe pas de sommet $c_i \in Ch_1$ tel que $|N(c_i)| > \lceil \frac{1}{3}|C| \rceil + 1$ alors on peut colorer $N(Ch_1)$ de la façon suivante. On commence par colorer, avec la couleur 1, puis 2 et 3, $N(c_1) \backslash N(c_2)$, puis $N(c_1) \cap N(c_2)$, $N(c_2) \backslash (N(c_1) \cup N(c_3))$, ... , $N(c_m) \backslash N(c_{m-1})$. Comme $|N(c_i)| \leq \lceil \frac{1}{3}|C| \rceil + 1$, il ne peut pas y avoir deux changements de couleur dans le voisinage d'un même sommet c_i. De plus, on a épuisé au moins deux couleurs pour colorer Ch_1, donc on a une 3-coloration acceptable de C dans G.

2eme sous-cas : Supposons qu'il existe un sommet $c_i \in Ch_1$ tel que $|N(c_i)| > \lceil \frac{1}{3}|C| \rceil + 1$. Il peut exister au plus 4 sommets $c_i \in Ch_1$ tels que $|N(c_i)| > \lceil \frac{1}{3}|C| \rceil + 1$. En effet, un sommet de ce type recouvre au moins $\lceil \frac{1}{3}|C| \rceil + 2$ sommets de C ; deux sommets en recouvrent au moins $\lceil \frac{1}{3}|C| \rceil + 3$; trois sommets en recouvrent au moins $\lceil \frac{2}{3}|C| \rceil + 4$; quatre sommets en recouvrent au moins $\lceil \frac{2}{3}|C| \rceil + 5$ et cinq sommets en recouvrent au moins $|C| + 6$ ce qui est impossible. On essaye de colorer $N(Ch_1)$ de la façon suivante. On commence par colorer avec la couleur 1, puis 2 et 3, $N(c_1) \backslash N(c_2)$, puis $N(c_1) \cap N(c_2)$, $N(c_2) \backslash (N(c_1) \cup N(c_3))$, ... , $N(c_m) \backslash N(c_{m-1})$. Supposons que cette coloration ne soit pas acceptable car il y a deux changements de couleur dans le voisinage d'un même sommet. Soit c_i le sommet le plus à gauche pour lequel la coloration n'est pas acceptable. c_i n'est pas c_1 car sinon pour avoir les deux changements de couleur dans son voisinage, il faudrait que $|N(c_i)| > \lceil \frac{2}{3}|C| \rceil$.



Soient k la distance entre la frontière gauche de c_i et le changement de couleur a et k' la distance entre la frontière droite de c_i et le changement de couleur b. On a $1 \leq k \leq \lceil \frac{1}{3}|C| \rceil - k'$ et $1 \leq k' \leq \lceil \frac{1}{3}|C| \rceil - k$, car $|N(c_i)| \leq \lceil \frac{2}{3}|C| \rceil$ et $|N(c_i)| \geq k + \lceil \frac{1}{3}|C| \rceil + k'$. Si on décale la coloration vers la gauche de k pas, en colorant à droite avec la couleur 3 puis la couleur 1, on va introduire au plus un nouveau changement de couleur c. a coincide avec la frontière gauche de c_i donc c_i ne contient plus que le changement de couleur b et il accepte la coloration. De plus seul l'un des pseudo-voisins de c_i peut contenir b. Supposons que c_{i+1} contienne b et c, cela signifie que $|N(c_i) \cup N(c_{i+1})| > \lceil \frac{2}{3}|C| \rceil$. Comme il n'y a pas de conflit, on a donc $|N(c_i) \cap N(c_{i+1})| \leq \lceil \frac{1}{3}|C| \rceil$. Si on décale encore la coloration vers la gauche afin de faire coincider b et la frontière gauche de c_{i+1}, on est sûr que c_i est le seul sommet qui contient b et qu'il ne contient pas c car $|N(c_i) \cap N(c_{i+1})| \leq \lceil \frac{1}{3}|C| \rceil$. Supposons que c_{i-1} contienne a et b, cela signifie que $|N(c_i) \cap N(c_{i-1})| > \lceil \frac{1}{3}|C| \rceil$ donc $|N(c_i) \cup N(c_{i-1})| \leq \lceil \frac{2}{3}|C| \rceil$. D'où si l'on décale encore pour faire coincider a et la frontière gauche de c_{i-1}, alors c_i et c_{i+1} sont les seuls à contenir b et ils ne contiennent ni a ni c. On a donc une coloration acceptable de C pour Ch_1 qui épuise au moins la couleur 2.

CQFD (Ouf, c'est pas trop tôt !)

■

Proposition 5 *Si I est un cycle de pseudo-voisins et qu'aucun couple de ses sommets n'est en conflit alors il existe une 3-coloration acceptable de C dans G.*

Preuve :

Il suffit de reprendre la méthode pour construire une coloration acceptable dans le 2eme sous-cas du 2eme cas de la preuve de la proposition 4.

■

Théorème 6 *L'algorithme 1 décide si la largeur de branche d'un splitgraphe de cordes $G = (I, C, E)$ est $\lceil \frac{2}{3}|C| \rceil$.*

De plus, sa complexité est $O(n + m)$ où n est le nombre de sommets du splitgraphe et m son nombre d'arêtes.

Preuve :

Les propositions 4 et 5 nous montrent qu'il suffit de garder les sommets de I qui vérifient les conditions du lemme 3 puis de tester s'il existe un sommet $i \in I$ tel que $|N(i)| > \lceil \frac{2}{3}|C| \rceil$ ou s'il existe un couple de sommets de I qui sont en conflit. Pour trouver les sommets maximaux pour l'inclusion ou tester si leurs voisinages sont trop petits ou trop grands, on peut le faire en parcourant une seule fois les sommets de l'indépendant ; cette étape se fait en $O(n + m)$. Sur les sommets restants de l'indépendant, on fait un nouveau parcours pour construire le graphe de pseudo-voisinage. On vérifie en même temps si deux pseudo-voisins ne sont pas en conflit ; cette étape aussi se fait en $O(n + m)$. Au final, on a un algorithme en $O(n + m)$.

■

Algorithme 1 `splits` : Algorithme pour les splitgraphes de corde

entrée:
$G = (I, C, E)$ un splitgraphe de cordes
sortie:
vrai : si la largeur de branche de G est $\lceil \frac{2}{3}|C| \rceil$

début
result $\leftarrow$ VRAI
pour chaque $i \in I$ **faire**
si $N(i)$ n'est pas maximal **alors**
$I \leftarrow I \backslash \{i\}$
si $|N(i)| > \lceil \frac{2}{3}|C| \rceil$ **alors**
result$\leftarrow$ FAUX
pour chaque i, $j \in I$ pseudo-voisins **faire**
si $|N(i) \cap N(j)| > \lceil \frac{1}{3}|C| \rceil$ et $|N(i) \cup N(j)| > \lceil \frac{2}{3}|C| \rceil$ **alors**
result$\leftarrow$ FAUX
rendre (result)
fin

0.3 Décompositions parallèles

Pour démontrer l'un des lemmes qui suit, il me faut d'abord présenter les notions de décompositions en branche parallèles et de profil d'une décomposition introduites par Frédéric Mazoit dans [8].

Définition 19 (frontière essentielle) *Soient $\Theta = (T, \tau)$ la décomposition en branches d'un graphe G et S la frontière de la séparation associée à une arête de T. S est* essentielle *pour Θ s'il n'existe pas d'autre frontière de Θ contenant S.*

Définition 20 (profil d'un graphe) *Le* profil *de Θ est le vecteur $(u_n, \ldots, u_1)$ où u_i est le nombre de frontières essentielles de taille i.*

Les profils des décompositions en branches de G sont ordonnés selon l'ordre lexicographique : $(u_n, \ldots, u_1) < (v_n, \ldots, v_1)$ s'il existe $i \in [1..n]$ tel que $u_i < v_i$ et $u_j = v_j$ pour $i < j \leq n$. Le profil d'un graphe *G est le profil minimum des décompositions en branches de G. Une* décomposition en branches optimale *est une décomposition en branches dont le profil est minimum. Clairement, une décomposition en branches optimale de G est de largeur $\beta(G)$.*

Propriété 4 ([8]) *Soient G un graphe et H une triangulation de G correspondant à une décomposition en branches optimale. Alors G et H ont le même profil.*

Définition 21 (décomposition parallèle) *Un décomposition en branches dont les frontières ne se croisent pas est une* décomposition parallèle.

Théorème 7 ([8]) *Soit $G = (V, E)$ un graphe. Une décomposition en branches optimale Θ de G n'a pas de frontières qui se croisent.*

Propriété 5 ([8]) *Soient $G = (V, E)$ un graphe et $\Theta = (T, \tau)$ une décomposition en branches optimale de G. Les séparateurs minimaux de la triangulation G_Θ de G sont des séparateurs minimaux de G.*

0.4 Méthode de Kloks

Dans cette section, je tente d'appliquer à la largeur de branche la méthode utilisée par Kloks pour calculer la largeur arborescente des graphes de cordes. Cette méthode consiste à considérer le polygone convexe défini par l'ensemble de points Z défini ci-après et à trouver une triangulation plane de poids minimal de ce polygone, le poids du polygone étant défini comme le nombre maximum de cordes qui croisent un même triangle.

Le théorème 4 montre qu'une triangulation minimale de G est caractérisée par une famille de séparateurs minimaux de G. Dans le cas des graphes de cordes, ces séparateurs minimaux ont une interprétation géométrique simple. Pour cela, on introduit les notions de points et de cordes de contrôle.

Définition 22 (points et cordes de contrôle, coupe) *Etant donné un graphe de cordes à n sommets, on définit l'ensemble Z de ses* points de contrôle *en plaçant un point de contrôle entre tous les couples d'extrémités consécutives de cordes sur le cercle. Il y a exactement $2n$ points de contrôle. Une* corde de contrôle *est une corde du cercle dont les extrémités sont des points de contrôle. Soit s une corde de contrôle, la* coupe *de s, notée $C(s)$, est l'ensemble des sommets de G dont les cordes croisent s.*

Définition 23 *Deux cordes de contrôle se croisent si elles ont une intersection non-vide mais n'ont pas d'extrémités communes.*

Définition 24 *Soient deux cordes s_1 et s_2 qui ne s'intersectent pas. Une corde de contrôle s est* entre *s_1 et s_2 si tout chemin d'une extrémité de s_1 vers une extrémité de s_2 sur le cercle passe par une extrémité de s.*

Si a et b sont deux sommets non-adjacents d'un graphe de cordes, alors les cordes correspondantes ne s'intersectent pas dans le modèle circulaire. Si on prend une corde de contrôle s entre les cordes de a et b, on a clairement $C(s)$ qui est un séparateur de a et b. Le lemme qui suit nous montre que les cordes de contrôle nous fournissent une représentation simple des séparateurs minimaux.

Lemme 6 ([10]) *Soient G un graphe de cordes et a, b deux sommets non adjacents de G. Pour tout a, b-séparateur minimal S, il existe une corde de contrôle s* entre *a et b telle que les cordes correspondant aux sommets de S sont exactement celles qui croisent s ($C(s) = S$).*

On va chercher à découper le graphe en « morceaux » à l'aide de cordes de contrôle, où chaque « morceau » sera une clique maximale d'une triangulation H correspondant à une décomposition en branches de profil minimal.

Définition 25 (composant, réaliseur) *Soit $Y \subseteq Z$ un ensemble de sommets de contrôle avec au moins trois éléments. Considérons le polygone convexe $P(Y)$ ayant Y pour ensemble de sommets. Le* composant *$G(Y)$ est le sous-graphe de G induit par l'ensemble des sommets correspondant à des cordes qui intersectent $P(Y)$. Le* réaliseur *$R(Y)$ du composant $G(Y)$ est le graphe $G(Y)$ auquel on a rajouté les arêtes nécessaires pour que tous les couples de sommets dont les cordes traversent un même côté du polygone $P(Y)$ soient adjacents.*

En conséquence, chaque composant est un sous-graphe de son réaliseur et $G = G(Z) = R(Z)$. De plus chaque corde correspondant à un sommet de $G(Y)$ intersecte exactement deux côtés de $P(Y)$ et donc si Y est un ensemble de trois sommets de contrôle, $R(Y)$ est une clique.

Les réaliseurs vont nous permettre de représenter ces « morceaux ». Il faut se souvenir que les séparateurs minimaux d'un graphe triangulé sont des cliques. Par chance, on a les deux lemmes qui suivent nous permettant de travailler uniquement sur le modèle circulaire du graphe initial.

Lemme 7 ([10]) *Si $G(Y)$ est un composant alors son réaliseur $R(Y)$ est un graphe de cordes.*

Définition 26 (Y-corde de contrôle) *Soient $G(Y)$ un composant et $R(Y)$ son réaliseur. Une corde de contrôle s du modèle circulaire de $R(Y)$ est une Y-corde de contrôle si ses extrémités sont des éléments de Y.*

Lemme 8 ([10]) *Soient $R(Y)$ le réaliseur d'un composant $G(Y)$, a, b deux sommets non-adjacents dans $R(Y)$ et S un a,b-séparateur minimal dans $R(Y)$. Alors il existe une Y-corde de contrôle s telle que $S = C(s)$.*

On essaie à présent de corréler les largeurs de branche du graphe entier et des « morceaux ».

Définition 27 (k-possible) *Un composant $G(Y)$ est k-possible si le réaliseur $R(Y)$ a une largeur de branche au plus k.*

Lemme 9 *Soit Y un ensemble de sommets de contrôle avec au moins quatre éléments tels que que le composant $G(Y)$ a au moins $\lceil \frac{3k+1}{2} \rceil$ sommets. On a alors la propriété suivante :*
Si $G(Y)$ est k-possible alors il existe une Y-corde de contrôle s avec $|C(s)| \leq k$ et divisant le polygone $P(Y)$ en deux nouveaux polygones avec comme ensembles de sommets Y_1 et Y_2 tels que les composants $G(Y_1)$ et $G(Y_2)$ aient tous les deux moins de sommets que $G(Y)$ et tels que $G(Y_1)$ et $G(Y_2)$ soient k-possibles.

Preuve :

Supposons que $G(Y)$ soit k-possible. Considérons une triangulation H correspondant à une décomposition en branche optimale de $R(Y)$, on a $\beta(H) = \beta(R(Y)) \leq k$. Puisque le nombre de sommets est au moins $3k + 1/2$, il doit y avoir au moins deux cliques maximales distinctes dans H. Puisque G est connexe, $G(Y)$, $R(Y)$ et H le sont aussi ; donc il existe deux cliques maximales Ω et Ω' de H non disjointes. De plus $S = \Omega \cap \Omega'$ est un séparateur minimal de H donc de $R(Y)$ d'après la propriété 5 et il est de taille au plus k car $G(Y)$ est k-possible. Soient a, b deux sommets de $R(Y)$ pour lesquels S est un séparateur minimal. D'après le lemme 8, il existe une Y-corde de contrôle s correspondant à S située *entre* les cordes correspondant à a et b. s divise le polygone $P(Y)$ en deux nouveaux polygones. Soient Y_1 et Y_2 les sommets de ces deux polygones. On peut supposer que $G(Y_1)$ contient la corde correspondant à a et $G(Y_2)$ celle correspondant à b. Soit C_a et C_b les sommets de $G(Y_1)$ et $G(Y_2)$ respectivement. Comme S induit une clique dans H, on en déduit que les sous-graphes induits $H[C_a]$ et $H[C_b]$ sont des triangulations de $R(Y_1)$ et $R(Y_2)$. De plus $R(Y_1)$ et $R(Y_2)$ sont des mineurs de $H[C_a]$ et $H[C_b]$ qui sont des mineurs de H donc d'après le lemme 1, $\beta(R(Y_1)) \leq \beta(H[C_a]) \leq k$ et $\beta(R(Y_2)) \leq \beta(H[C_b]) \leq k$ et donc $G(Y_1)$ et $G(Y_2)$ sont k-possibles. Puisque a n'est pas dans $G(Y_2)$ et b n'est pas dans $G(Y_1)$, il suit que ces deux composants ont moins de sommets que $G(Y)$.

■

Lemme 10 *La réciproque du lemme 9 à savoir :*

S'il existe une Y-corde de contrôle s vérifiant $|C(s)| \leq k$ et divisant le polygone $P(Y)$ en deux nouveaux polygones plus petits avec comme ensembles de sommets Y_1 et Y_2 tels que les composants $G(Y_1)$ et $G(Y_2)$ sont tous deux k-possibles, alors $G(Y)$ est aussi k-possible.

est fausse.

Preuve :

Voici un contre-exemple de cette réciproque.
Considérons le splitgraphe suivant :

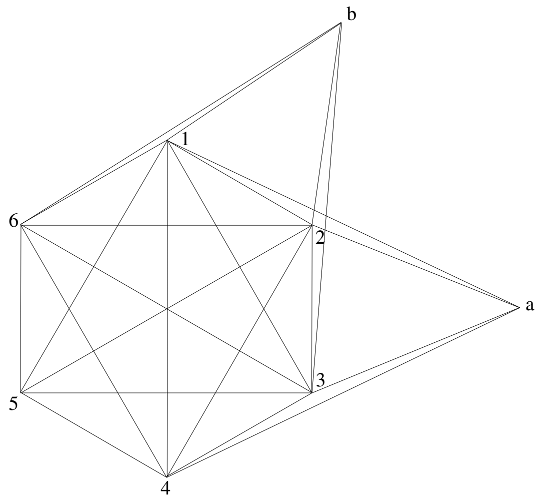
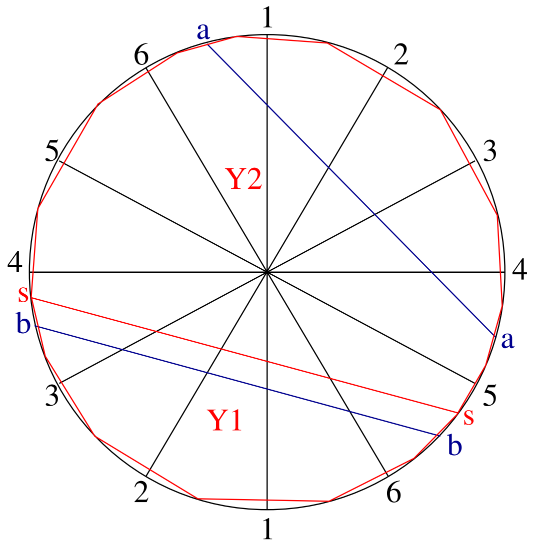


FIG. 5 – Un contre-exemple

Ce splitgraphe n'est pas de largeur de branche 4 car il n'admet pas de 3-coloration acceptable de l'ensemble des sommets de la clique. En effet, les 6 sommets doivent être colorés avec 3 couleurs et chaque couleur doit être présente sur au plus deux sommets. Or, les sommets 1,2 et 3 sont inclus dans les voisinages de a et de b ; donc ils doivent être colorés avec deux couleurs. Supposons sans perte de généralité que deux de ces 3 sommets sont colorés par la couleur 1 et le troisième par la couleur 2. Dès lors, pour que la coloration soit acceptable pour le sommet a (respectivement b) il faut que le sommet 4 (respectivement 6) soit coloré avec la couleur 2. Mais on ne peut attribuer la couleur 2 qu'à un seul sommet supplémentaire, il n'existe donc pas de 3-coloration acceptable.
Ce splitgraphe est un graphe de cordes et il peut être séparé par une corde de contrôle de poids 4 en deux composants comportant strictement moins de sommets comme le montre le dessin suivant.



On peut facilement constater que $G(Y_1) = R(Y_1) = K_4$ qui est de largeur de branche 3. De plus $G(Y_2) = R(Y_2)$ et $R(Y_2)$ est un splitgraphe dont la clique est de taille 6 qui admet une 3-coloration de manière évidente et donc est de largeur de branche 4. D'où $G(Y_1)$ et $G(Y_2)$ sont tous deux 4-possibles alors que $G(Z) = R(Z) = G$ ne l'est pas, ce qui contredit la réciproque du lemme 9.

$\blacksquare$

Définition 28 (pseudo-triangulation) *Soit P un polygone avec m sommets. Une* pseudo-triangulation *de P est un ensemble d'au plus $m-3$ diagonales de P qui ne se croisent pas. Elles divisent l'intérieur de P en au plus $m-2$ polygones.*

Définition 29 (poids) *Soit Y un ensemble d'au moins trois sommets de contrôle. Considérons une pseudo-triangulation T de $P(Y)$. Le* poids *d'un polygone est le cardinal maximum d'une de ses frontières. Le* poids *de la pseudo-triangulation, $w(T)$, est le poids maximum d'un sous-polygone.*

Théorème 8 *Soit Y un ensemble d'au moins 3 sommets de contrôle. Un composant $G(Y)$ n'est k-possible que s'il existe une pseudo-triangulation T de $P(Y)$ de poids au plus k et où chaque sous-polygone a au plus $\lceil \frac{3k+1}{2} \rceil$ sommets.*

Preuve :

Supposons que $G(Y)$ soit k-possible. Si $G(Y)$ a au plus $\lceil \frac{3k+1}{2} \rceil$ sommets alors la pseudo-triangulation de $P(Y)$ ne comportant aucune diagonale est satisfaisante. Procédons par induction sur le nombre de sommets de $G(Y)$. Supposons que $G(Y)$ ait plus que $\lceil \frac{3k+1}{2} \rceil$ sommets alors Y est de taille au moins 4 car le réaliseur d'un triangle est une clique et donc $G(Y)$ ne pourrait pas être k-possible. D'après le lemme 9, il existe une Y-corde de contrôle qui divise le polygone $P(Y)$ en deux nouveaux polygones $P(Y_1)$ et $P(Y_2)$ tels que les composants $G(Y_1)$ et $G(Y_2)$ soient k-possibles et aient tous les deux moins de sommets que $G(Y)$. Par induction, il existe des pseudo-triangulations T_1 et T_2 de $P(Y_1)$ et $P(Y_2)$ de poids au plus k et où chaque sous-polygone a au plus $\lceil \frac{3k}{2} \rceil$ sommets de G. Alors on a clairement $T = T_1 \cup T_2$ qui est une pseudo-triangulation de $P(Y)$ satisfaisante.

$\blacksquare$

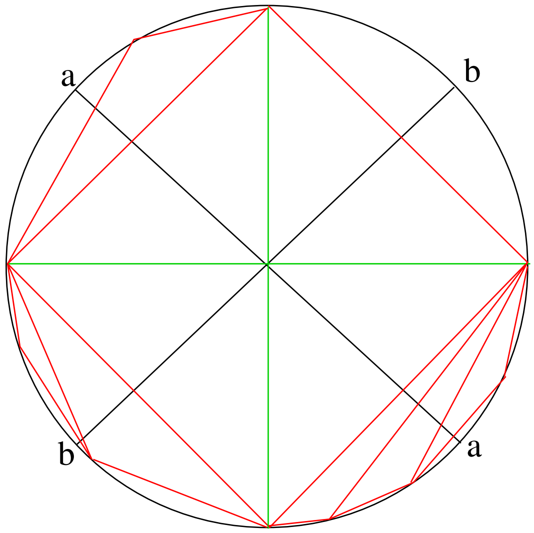
Corollaire 3 *Soient G un graphe de cordes et Z l'ensemble des points de contrôle d'un modèle circulaire de G. Alors $\beta(G)$ est strictement supérieure au plus grand k tel qu'il n'existe pas de pseudo-triangulation de $P(Z)$ de poids au plus k et où chaque sous-polygone a au plus $\lceil \frac{3k}{2} \rceil$ sommets de G.*

La démarche de Kloks ne nous permet pas d'obtenir une égalité entre la largeur de branche d'un graphe de corde et le poids d'une pseudo-triangulation minimale de son modèle circulaire. Néanmoins, une triangulation de poids minimal nous fournit une borne supérieure de la largeur de branche.

Théorème 9 *Soient G un graphe de cordes, $M(G)$ un modèle circulaire de G et Z l'ensemble des points de contrôle de $M(G)$. $\beta(G)$ est inférieure ou égale au poids de toute triangulation plane de $P(Z)$.*

Preuve :

Il faut remarquer que pour tout couple de cordes de $M(G)$ qui se croisent, il existe forcément une corde de contrôle de la triangulation plane qui croise ces deux cordes. Sur l'exemple qui suit on se rend bien compte qu'il faut que l'une des deux cordes de contrôle vertes soit ajoutée aux rouges pour obtenir une triangulation plane.



Toute arête de G est donc incluse dans la coupe d'une corde de la triangulation plane. On associe à chaque triangulation plane le graphe T ayant pour sommets les triangles de la triangulation plane, et pour arêtes celles qui sont entre deux sommets correspondant à des triangles ayant un côté commun. Les arêtes sont en bijection avec les cordes de la triangulation plane. De plus, T est un arbre dont les noeuds internes sont de degré au plus 3. Supposons que les noeuds de T soient étiquetés par l'ensemble des sommets de G dont les cordes coupent les triangles correspondants. Si l'on associe à chaque arête e de T l'ensemble X_e des sommets de G qui étiquettent au moins un sommet des deux composantes connexes de $T \backslash e$, alors X_e est égal à $C(s)$ où s est la corde de contrôle associée à e. T est en fait un squelette de décomposition en branches. On obtient une décomposition en branches de G de largeur inférieure ou égale au poids de la triangulation plane comme suit. Tout d'abord, pour chaque arête f de G qui n'étiquette pas au moins une feuille de T, on va ajouter une feuille lui correspondant. Comme chaque arête f de G est incluse dans X_e pour une certaine e, on ajoute un sommet de degré

3 relié à une feuille étiquetée par f au milieu de e. Cette modification remplace e par 3 arêtes i, j, k et on a $X_i, X_j, X_k \subseteq X_e$. Un fois qu'on a rajouté toutes les arêtes de G dans des feuilles de T, on supprime les feuilles qui ne correspondent pas à des arêtes de G puis on ne garde qu'une feuille par arête. Enfin, on supprime les noeuds internes de degré 2 et on met pour chacun une arête entre ses deux voisins. On obtient ainsi une décomposition en branches de largeur inférieure ou égale au poids de la triangulation plane. ■

De plus, la borne supérieure obtenue pour la largeur de branche en regardant une triangulation plane de poids minimal est meilleure ou égale à celle obtenue en calculant la largeur arborescente et en utilisant le théorème 5 qui nous dit que :

$$\beta(G) \leq \alpha(G) + 1$$

En effet, la largeur arborescente est obtenue en retranchant 1 au nombre maximal de cordes qui coupent chaque triangle d'une triangulation minimale pour ce paramètre. Pour chaque triangle, le nombre de cordes qui croisent une de ses frontières est inférieur au nombre total de cordes qui le croisent. Donc si on optimise les triangulations pour minimiser le nombre maximal de cordes croisant une corde de contrôle de la triangulation, on obtiendra une borne supérieure $\mathcal{B}$ de $\beta(G)$ telle que $\mathcal{B} \leq \alpha(G) + 1$.

Le fait mentionné dans la preuve du théorème précédent, à savoir que toute triangulation plane d'un graphe de corde comportant une arête est au moins de poids 2, nous montre qu'il n'existe pas de triangulation plane ayant un poids égal à la largeur de branche du graphe pour tous les graphes de largeur de branche 0 ou 1 (cf lemme 2).

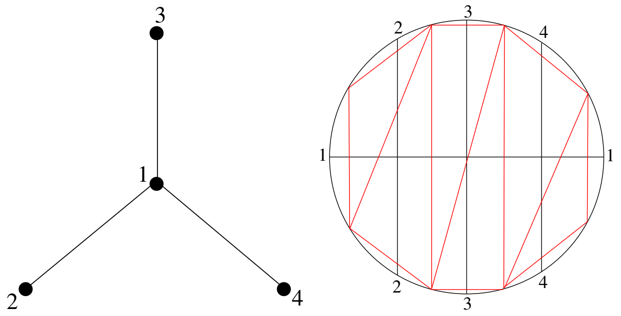


FIG. 6 – Un graphe de cordes de largeur arborescente 1 et une triangulation plane minimale de son modèle circulaire

En revanche, je n'ai pas réussi à trouver d'exemple de graphe de cordes ayant une largeur de branche supérieure ou égale à 2 et dont la triangulation plane minimale de $P(Z)$ soit de poids strictement supérieur. Il est intéressant de noter que les résultats relatifs à la largeur de branche d'un hypergraphe H dans l'article [15] font intervenir une relation minmax non pas directement entre la largeur de branche $\beta(H)$ et le nombre d'enchevêtrement $\theta(H)$ mais entre $\max(\beta(H), \gamma(H))$ et $\theta(H)$ où $\gamma(H)$ est la valence maximale d'une hyperarête de H. $\gamma(H)$ est donc égale à 2 dans le cas d'un graphe comportant au moins une arête autre qu'une boucle.

0.5 Méthode de Mazoit

Bouchitté et Todinca [2, 3] ont introduit la notion de clique maximale potentielle pour désigner les cliques maximales d'une triangulation d'un graphe associée à une décomposition arborescente de largeur minimum. Ils ont montré que si le nombre de séparateurs minimaux est polynomial en la taille du graphe, alors on peut lister en temps polynomial les cliques maximales potentielles et en extraire un ensemble formant les cliques maximales d'une triangulation minimale. Dans ce cas, le calcul de la largeur arborescente est polynomial. Mazoit [8] a étendu cette idée en introduisant la notion de branch-clique pour désigner les cliques maximales d'une triangulation associée à une décomposition en branches parallèle.

Définition 30 (séparateurs voisins, partie entre) *Un ensemble $\mathcal{S}$ de séparateurs minimaux est un ensemble de* séparateurs voisins *si pour tout $S \in \mathcal{S}$, il existe une composante connexe $C(S)$ de $G \backslash S$ telle que tout $S' \in \mathcal{S}$ est inclus dans $B(S) = S \cup C(S)$. On définit la* partie entre *$\mathcal{S}$ par $P(\mathcal{S}) = \bigcap_{\mathcal{S} \in \mathcal{S}} \mathcal{B}(\mathcal{S})$. On dira que $\mathcal{S}$ borde $P(\mathcal{S})$.*

Mazoit a montré qu'une branch-clique Ω d'un graphe G est la partie entre un ensemble de séparateurs voisins $\mathcal{S}$ de G. En définissant la largeur de branche d'une branch-clique Ω comme la largeur de branche de la clique $H[\Omega]$ à laquelle on a rajouté les hyperarêtes $e_S = \{S\}, S \in \mathcal{S}$, il obtient le théorème suivant.

Théorème 10 *Soit $G = (V, E)$ un graphe, si l'on peut calculer en temps polynomial une famille $\mathcal{F}$ de branch-cliques telle que pour toute $\Omega \in \mathcal{F}$, on peut calculer en temps polynomial la largeur de branche de Ω et telle qu'il existe une triangulation H de G avec $\beta(H) = \beta(G)$ et dont les cliques maximales sont toutes dans $\mathcal{F}$, alors la largeur de branche de G peut être calculée en temps polynomial.*

Il montre ensuite que pour les graphes d'intervalles circulaires, toute branch-clique peut être définie comme la partie entre au plus trois coupes associées à des cordes de contrôle qui ne se croisent pas et il obtient donc un nombre polynomial de branch-cliques dont la largeur de branches est calculable en temps polynomial. Ce faisant, il obtient l'appartenance à P du calcul de la largeur de branches des graphes d'intervalles circulaires. J'ai cherché à obtenir un résultat similaire pour les graphes de cordes.

Proposition 6 *Soient $\Theta = (T, \tau)$ une décomposition en branches de profil minimum de G un graphe de cordes connexe et Ω une clique maximale de la triangulation correspondante G_Θ bordée par la famille de séparateurs voisins $\mathcal{S}$. Soient e_1, e_2 et e_3 les arêtes de T incidentes à un sommet correspondant à Ω. Il existe un ensemble $\mathcal{L}$ de cordes de contrôle qui ne se croisent pas tel que $\bigcup_{l \in \mathcal{L}} C(l) = \bigcup_{S \in \mathcal{S}} S$ et pour tout $l \in \mathcal{L}$ il existe $e_i \in \{e_1, e_2, e_3\}$ telle que $C(l) \subseteq F(e_i)$. De plus, les cordes de contrôle de $\mathcal{L}$ définissent un polygone convexe et Ω est égale à l'ensemble des sommets dont les cordes intersectent ce polygone.*

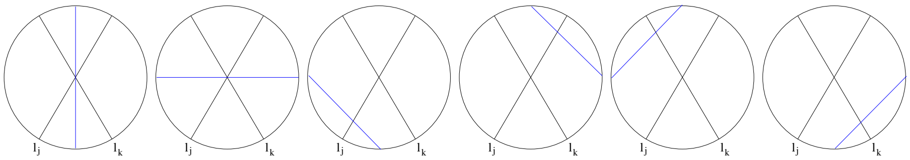
Preuve :

Pour $S_j \in \mathcal{S}$, par le lemme 6, il existe une corde de contrôle l_j entre u_j et v_j telle que $S_j = C(l_j)$.

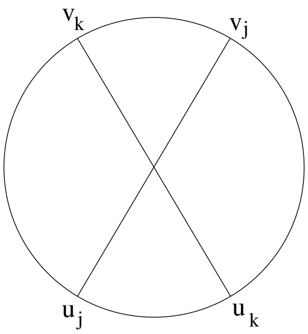
Chaque S_j est inclus dans l'une des $F(e_i)$. En effet, chaque S_j est un séparateur minimal de G_Θ qui borde Ω donc il existe une clique maximale Ω' de G_Θ telle que $S_j = \Omega \cap \Omega'$. D'où toutes les frontières associées aux arêtes de T qui sont sur un chemin entre un sommet correspondant à Ω et un sommet correspondant à Ω' contiennent S_j. Comme au moins une des arêtes e_1, e_2, e_3 fait partie d'un de ces chemins, on a $S_j \subseteq F(e_i)$, pour un certain i.

Cela implique qu'il existe au moins un ensemble de cordes de contrôle $\mathcal{L} = \{l_j / S_j \in \mathcal{S}\}$ tel que $\bigcup_{l \in \mathcal{L}} C(l) = \bigcup_{S \in \mathcal{S}} S$ et pour tout $l \in \mathcal{L}$ il existe $e_i \in \{e_1, e_2, e_3\}$ telle que $C(l) \subseteq F(e_i)$. Choisissons un tel ensemble $\mathcal{L}$ avec aussi peu d'éléments que possible.

Montrons que deux cordes distinctes l_j et l_k de $\mathcal{L}$ peuvent seulement se rencontrer à une extrémité. l_j et l_k correspondent à des éléments S_j et S_k de $\mathcal{S}$ donc $S_j||S_k$. Supposons que l_j et l_k s'intersectent en un point autre qu'une de leurs extrémités. Les cordes de M(G) qui croisent au moins une des deux sont de 6 types :

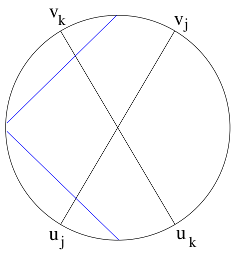


Les cordes des deux premiers types croisent l_j et l_k. Les cordes de type 3 et 4 croisent uniquement l_j. Les cordes de type 5 et 6 croisent uniquement l_k. On peut remarquer qu'on ne peut avoir en même temps de cordes du type 3 et 4 (respectivement 5 et 6) car sinon S_j (respectivement S_k) contient des sommets qui sont dans deux composantes connexes distinctes de $G\backslash S_k$ (respectivement $G\backslash S_j$), ce qui contredit $S_j||S_k$. S'il n'existe que des sommets de type 1 et/ou 2 alors $S_j = S_k$. S'il n'existe que des sommets de type 1,2,3 ; 1,2,4 ; 1,2,5 ; 1,2,6 alors $S_k \subseteq S_j$; $S_k \subseteq S_j$; $S_j \subseteq S_k$; $S_j \subseteq S_k$. Dans tous ces cas, on aurait pu se passer de l'une des cordes l_j ou l_k ce qui contredit la minimalité de $\mathcal{L}$. On suppose que G est connexe et l_j,l_k se croisent :

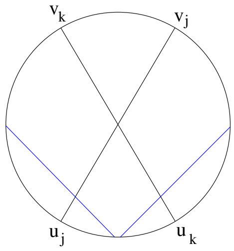


avec $u_j \neq u_k, u_j \neq v_k$ et $v_j \neq u_k, v_j \neq v_k$. u_j, v_j, u_k, v_k sont des points de contrôle donc il doit y avoir au moins une extrémité d'une corde entre u_j et u_k, v_j et v_k, u_j et v_k, v_j et u_k. Il nous reste 4 cas à étudier :

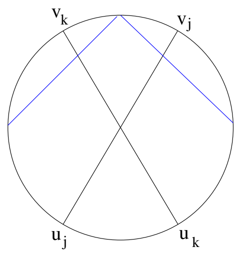
- cas 1 : cordes de types 3 et 5



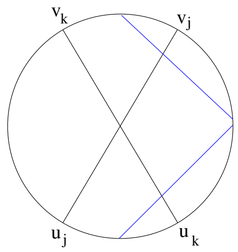
- cas 2 : cordes de types 3 et 6



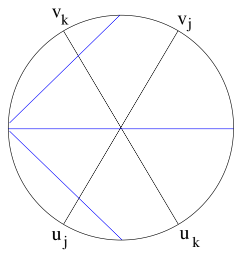
- cas 3 : cordes de types 4 et 5



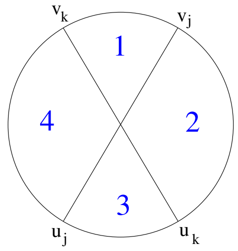
– cas 4 : cordes de types 4 et 6

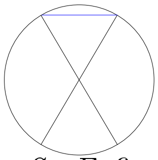
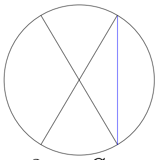
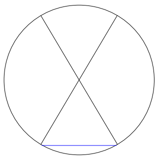


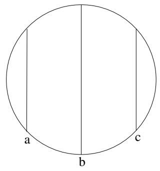
Comme G est connexe, il doit exister une corde du type 1 dans les cas 2 et 3 et une corde du type 2 dans les cas 1 et 4. Tous les cas étant symétriques, on se limite à l'étude du cas 1 :



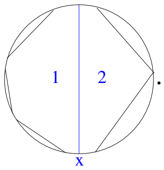
Pour toute corde l_j, Ω est un sous-ensemble d'un bloc de S_j donc comme G est connexe un bloc de S_j est inclus dans l'une des deux moitiés de cercle définies par l_j. On en déduit que Ω est un sous-ensemble des sommets correspondants aux cordes qui intersectent l'une des 4 zones suivantes :



Comme les sommets de S_j et S_k font partie de Ω, Ω est forcément inclus dans la zone 4. Il n'existe pas de corde du type 4 donc l_k sépare Ω de la zone 1 mais la corde l_m :  aussi et $S_m \subset S_k$. De même, l_n :  sépare Ω de la zone 2 et $S_n \subset S_j \cap S_k$. Enfin, l_o :  sépare Ω de la zone 3 et $S_o \subset S_j$. Ceci contredit le fait que S_j et S_k soient des séparateurs minimaux pour un couple de sommets dans $\Omega \times (G\backslash\Omega)$. On peut donc supposer que 2 cordes l_j et l_k de $\mathcal{L}$ peuvent seulement se rencontrer à leurs extrémités.

Montrons qu'il n'existe pas trois cordes a, b, c de $\mathcal{L}$ disposées comme suit : 

. S_a, S_b et S_c sont des séparateurs voisins donc $S_b, S_c \subseteq S_a \cup C_a$ où C_a est l'unique composante connexe de $G\backslash S_a$ qui intersecte $S_b \cup S_c$, $S_a, S_c \subseteq S_b \cup C_b$ où C_b est l'unique composante connexe de $G\backslash S_b$ qui intersecte $S_a \cup S_c$, $S_a, S_b \subseteq S_c \cup C_c$ où C_c est l'unique composante connexe de $G\backslash S_c$ qui intersecte $S_a \cup S_b$. De plus, les C_j ne peuvent être qu'un sous-ensemble des cordes situées d'un seul côté des cordes j. On ne peut donc pas avoir en même temps $S_a \cap C_b \neq \emptyset$ et $S_c \cap C_b \neq \emptyset$, d'où $S_a \subseteq S_b$ ou $S_c \subseteq S_b$. On pouvait donc se passer de la corde a ou de la corde c ce qui contredit la minimalité de $\mathcal{L}$.

On peut donc considérer que les cordes de $\mathcal{L}$ définissent un polygone convexe. Montrons qu'il n'existe pas de sommet $x \in G\backslash\Omega$ tel que la corde correspondante coupe ce polygone. Supposons le contraire : 

. On a les séparateurs minimaux S_i^1 qui correspondent à des cordes de contrôle dans la zone 1 et les séparateurs minimaux S_j^2 qui correspondent à des cordes de contrôle dans la zone 2. Soit S_l^k un de ces séparateurs minimaux. Tous les autres séparateurs minimaux sont inclus dans $S_l^k \cup C_l^k$ où C_l^k est une composante connexe de $G\backslash S_l^k$. Supposons qu'une corde d'un sommet de C_l^k croise la corde de x alors nécessairement $x \in C_l^k$. Puisque $x \notin \Omega$, il doit exister un séparateur $S_{l'}^{k'}$ tel que $x \notin C_{l'}^{k'}$. Donc, $\exists S_{l'}^{k'}$ tel que $\nexists y \in C_{l'}^{k'}$ dont la corde croise celle de x. D'où toutes les cordes de Ω qui croisent x sont incluses dans $S_{l'}^{k'}$. $C_{l'}^{k'}$ est donc située soit dans la zone 1 soit dans la zone 2 et donc tous les S_i^1 ou tous les S_j^2 sont inclus dans $S_{l'}^{k'}$ ce qui contredit la minimalité de $\mathcal{L}$. ■

Malheureusement, je n'arrive pas à borner par une constante le nombre de cordes de contrôle dans $\mathcal{L}$ et donc je n'arrive ni à donner une borne polynomiale au nombre de branch-cliques, ni à donner une borne polynomiale à la complexité du calcul de la largeur de branche d'une branch-clique.

0.6 Conclusion

J'ai tenté de savoir si le problème du calcul de la largeur de branche des graphes de cordes était polynomial ou s'il était NP-complet. Pour ce faire, j'ai repris trois méthodes qui avaient déjà été utilisées pour des problèmes très proches. Kloks et al ont montré que décider si la largeur de branche d'un splitgraphe (I, C, E) vaut $\lceil \frac{2}{3}|C| \rceil$ est NP-complet, mais pour les splitgraphes de cordes, j'ai trouvé un algorithme linéaire qui décide ce problème. On peut donc espérer trouver un algorithme polynomial pour calculer et non décider la largeur de branche des splitgraphes de cordes et par extension calculer la largeur de branche des graphes de cordes. En reprenant la méthode de Kloks, j'ai obtenu une borne supérieure de la largeur de branche ; de plus cette borne est plus précise que si je calcule la largeur arborescente du graphe et l'utilise avec le théorème 5. C'est la meilleure borne connue à ce jour pour les graphes de cordes et on a la preuve qu'elle n'est pas supérieure à $\frac{3}{2}$ de la valeur optimale. Cette méthode échoue car on n'arrive pas à prouver que l'on peut forcément trianguler certains réaliseurs sans augmenter le poids de la pseudo-triangulation. Cependant, si la méthode de coloration continue des splitgraphes de cordes se généralise pour calculer leur largeur de branche, on pourra sans doute trouver une méthode efficace de répartition des frontières des réaliseurs et donc on pourra achever la décomposition de façon optimale.

Plus généralement, on peut continuer à s'intéresser à la largeur de branche car ce domaine est peu étudié et il pourrait apporter des avancées importantes pour l'algorithmie des graphes. Plus précisément, j'aimerais poursuivre mon travail sur les graphes de cordes et les graphes d'intersection en général. L'étude d'heuristiques garanties pour le calcul de la largeur de branche est aussi un sujet qui m'intéresse. En effet, cela améliorerait les techniques de résolution basée sur les décompositions en branches des problèmes d'optimisation.

Bibliographie

[1] S. Arnborg, J. Lagergren, and D. Seese. Easy problems for tree-decomposable graphs. *Journal of Algorithms*, 12(2) :308–340, 1991.

[2] V. Bouchitté and I. Todinca. Minimal triangulations for graphs with "few" minimal separators. In *Proceedings 6th Annual European Symposium on Algorithms (ESA'98)*, volume 1461 of *Lecture Notes in Computer Science*, pages 344–355. Springer-Verlag, 1998.

[3] V. Bouchitté and I. Todinca. Listing all potential maximal cliques of a graph. In *Proceedings 17th Annual Symposium on Theoretical Aspects of Computer Science (STACS 2000)*, volume 1770 of *Lecture Notes in Computer Science*, pages 503–515. Springer-Verlag, 2000.

[4] W. Cook and P. Seymour. Tour merging via branch-decomposition, December 2002.

[5] B. Courcelle. The monadic second-order logic of graphs III : Treewidth, forbidden minors and complexity issues. *Informatique Théorique*, 26 :257–286, 1992.

[6] P. Damaschke. The Hamiltonian circuit problem for circle graphs is NP-complete. *Information Processing Letters*, 32 :1–2, 1989.

[7] F.Gavril. Algorithms on circle graphs. *Networks*, 4 :357–369, 1974.

[8] F.Mazoit. A general scheme for deciding the branchwidth. Technical Report RR2004-34, ENS Lyon, juin 2004.

[9] W.-L. Hsu. Maximum weight clique algorithms for circular-arc graphs and circle graphs. *SIAM J. Comput.*, 14 :224–231, 1985.

[10] T. Kloks. Treewidth of circle graphs. In *Proceedings 4th Annual International Symposium on Algorithms and Computation (ISAAC'93)*, volume 762 of *Lecture Notes in Computer Science*, pages 108–117. Springer-Verlag, 1993.

[11] T. Kloks, J. Kratochvíl, and H. Müller. New branchwidth territories. In *Proceedings 16th Annual Symposium on Theoretical Aspects of Computer Science (STACS'99)*, volume 1563 of *Lecture Notes in Computer Science*, pages 173–183. Springer-Verlag, 1999.

[12] G.L. Miller M.R. Garey, D.S. Johnson and C.H. Papadimitriou. The complexity of coloring circular arcs and chords. *SIAM Journal on Algebraic and Discrete Methods*, pages 216–227, 1980.

[13] A. Parra and P. Scheffler. Characterizations and algorithmic applications of chordal graph embeddings. *Discrete Appl. Math.*, 79(1-3) :171–188, 1997.

[14] N. Robertson and P.D. Seymour. Graphs minors. III. Planar tree-width. *J. of Combinatorial Theory Series B*, 36 :49–64, 1984.

[15] N. Robertson and P.D. Seymour. Graphs minors. X. Obstruction to tree-decomposition. *J. of Combinatorial Theory Series B*, 52 :153–190, 1991.

[16] T. Kashiwabara S. Masuda, K. Nakajima and T. Fujisawa. Efficient algorithms for finding maximum cliques of an overlap graph. *Networks*, 20 :157–171, 1990.

[17] J. Spinrad. Recognition of circle graphs. *J. Algorithms*, 16 :264–282, 1994.

[18] I. Todinca. *Aspects algorithmiques des triangulations minimales des graphes*. PhD thesis, École Normale Supérieure de Lyon, 1999.